Object Oriented Programming **FAST-NU** **Semester Project**
BS(DS), BS(AI) **Deptt. of DS & AI** **Due Date: 12 May 2023**

FAST Examination System

Instructions:

In this project you are required to design and implement an examination system which will enable the instructor to set, conduct, mark and result preparation. Moreover, additional analytics about the result will be compiled and shared with the instructor.

The system will be a console based application. There would be 2 kinds of users of the system, i.e. 1. Teacher, 2. Student. The teacher would create a question bank, then would create Quiz/Assignment and set the date/time for the quiz. Then after an evaluation is attempted by the student. The software would automatically mark the Quiz and generate marks report and another analytics report which will show some useful insights about the evaluation. The system will store the data permanently on the disk thru file reading/writing.

The system would implement the following data flow:

1. Users creation: The system will generate logins of the instructor and the students by reading their names, from the text files.
2. Course offering: Courses are offered as per the details given in the course offering list.
3. Course registration: The students will be registered in 5-7 courses randomly.
4. Question bank preparation by the instructor: Either enter the MCQs, True/False Questions, or subjective questions. The questions are divided into topics.
5. Quiz setting/generation: Quiz generation from the stored questions. The teacher selects the topics from where the quiz will be set. No. of questions from each topic and marks of each will be set by the teacher accordingly with date/time setting with maximum time limit.
6. Quiz conduct: The system will allow the registered students to attempt the quiz at the set time. The system will randomize the question sequence and answer options to increase security of the test. The quiz will be time bound, till the maximum time set by the instructor.
7. Answer Key generation: For each quiz the system will generate an answer key/solution from the stored answers.
8. Result preparation: After conclusion of the quiz the result will be prepared by the system. The marks sheet of the whole class is prepared and shown/written to the disk.
9. Quiz analytics for individual quiz and all quizzes of the course. In this section show the percentage of correct answers to each question during the test, this will enable the teacher to see a bar graph to guage the effectiveness of the questions, the bar will be low if none of the students get the question right, and it will be high in proportion to the number of students who get it right.
10. Generate attendance sheet of the quiz, i.e. Mark as absent the registered students who did not attempt the quiz.
11. The system must be able to handle multiple

System

- Data of the students and list of offered courses in a semester will be provided to you in the form of CSV files.
- There will be multiple menus to support the desired functionalities.
- Generate appropriate menus to get required input from the user. Deal with all invalid inputs and handle the exceptions if they occur. The program should never crash.

- Use Object oriented design to model the problem, like create classes of Teacher, student, Quiz, Question, Date, Time etc.

- Users are presented with a menu to navigate the entire system, with a subsequent process menu and a link to the main menu provided upon completion of each step. General information about the exam system is available to all users without requiring a login. However, registered users are the only ones who can access system-specific information such as settings/attempting the evaluation.
- Implement an authentication mechanism for the users to login, the users would set up their password, NU ID would be their login ID. When the users enter a password, the system will display asterisks (*). The users will have the option to change the password thru the appropriate menu. The password provided must have minimum of 6 length. The password must contain uppercase, lowercase, numeric digit, and special characters.

- There will be 2 phases of this project. 1. Design of the system, 2. Implementation of the system.
  - Design submission due date: 8 May 2023
  - Implementation submission due date: 12 May 2023.

General programming requirements:

- Use dynamic memory allocation, de-allocation, care must be maintained while dealing with dynamic memory to avoid memory leakages.
- The system must never crash and deal with the logical errors accordingly.
- Comment the code properly.
- Use recommended programming practices to write code.

Submission

1. Program Code with all requirements + Viva Demo	(70 marks)
2. Report	(20 marks)
3. Class Diagram (made on Microsoft Visio)	(10 marks)

Deadline: 12 May 2023

Create a report detailing the application of OOP concepts in the project, such as identifying the use of Inheritance, Aggregation, Association, Composition, and constant/static data members. Additionally, provide screenshots to support the explanation and include a class diagram that depicts the scenario. Also include screenshots of the system and write guidelines on how to use the system.

Submission instructions

- This is an individual project.
- Make a folder, and write your roll number as the folder name. Example: 22i-XXXX.
- Zip the folder that contains all your .cpp files with the same name as the folder name.
- Example: 20i-XXXX.zip. (Right Click folder -> Send To -> Compressed zipped folder)
- DO NOT compress the folder with .rar / .7zip extension.
- Report is not part of zip.
- Submit Class Diagram separately as well (original file + exported image).
- If any submission does not follow the guideline will be awarded 0 marks in the project.

General instructions

- Plagiarism will result in F-grade in this course.
- No late submission will be accepted so now is the right time to start working on it.
- Coding language: C++
- All coding standards must be followed (meaningful variable name, code comments etc.)
- Apply all validations for invalid inputs. Use Exception Handling.
- Use of Vectors and similar container is not allowed.

Good Luck